University of Sindh Journal of Information and Communication Technology (USJICT)
Volume 5, Issue 4, December 2021
ISSN-E: 2523-1235, ISSN-P: 2521-5582
Website: http://sujo.usindh.edu.pk/index.php/USJICT/

Forecasting Multan estate prices using optimized regression techniques

Adnan Akhter, Muhammad Bux Alvi, Majdah Alvi

Department of Computer Systems Engineering, The Islamia University of Bahawalpur, Pakistan
adnanking7744@gmail.com

Abstract: Purchasing a house or plot has become a complicated task for an average person due to budget constraints and market situation. An individual does not know the prices of the plots and gets trapped by middle man. This paper proposes a solution for this problem by predicting the plot prices using machine learning approach, leveraging Multiple Linear Regression, Gradient Boosting Regression, and Random Forest regression techniques. This work compares the performance of these three algorithms by hyper-parameter tuning using gird search, and random search for checking which one is adequate in terms of R^2 scores and error rates. Factors that influence the prices of the plots include plot covered area, physical condition of the plot, area, and population. Gradient boosting regression has surpassed all other machine learning methods, achieving the lowest error rates and highest R-squared score of 0.987 with grid search. The resultant predictive systems can help the folk in three ways. 1) safety from deception 2) budget oriented instant information, and 3) time saving.

Keywords: regression methods, machine learning, plot prediction, hyper parameter tuning

I. INTRODUCTION

Property buying and selling is an important economically factor. Buying or selling the plot, for an average person, is a difficult task because he/she does not know the price of the plot due to his/her limited market insight. People spend their lives saving money to buy a plot but unfortunately some of them get trapped by fraudulent. They purchase the plot at the higher cost in comparison to the actual price of the plot which is very low, in actual. Similarly, less aware people sell their plot at lower cost. The objective of this study is to solve the problem of the plot prices for the sellers/purchasers where they can easily assess actual price of the assets under consideration.

This paper proposes the solution of the problem by developing a system that can forecast the plot price. As plot prices are real values, therefore, multivariate regression techniques are better choice to apply for estimating the cost of a plot, given plot parameters. Three models are developed using multiple linear regression, gradient boosting regression, random forest regression and are tuned by hyper-parameters using grid search and random search for this work, using **Multan City Plotting** data-set. That predictive models are developed with the help of data-set splitting method. Finally, the performance metrics of these models are evaluated using R^2 score, MAE, MSE, and RMSE methods. These methods are employed to compare the model's performance without optimization and with optimization for checking which model is adequate in terms of R^2, and error values.

The figure 1 demonstrates the work that adopted to carry out for prediction without the optimization, starting from the selecting the data-set up to errors evaluation.

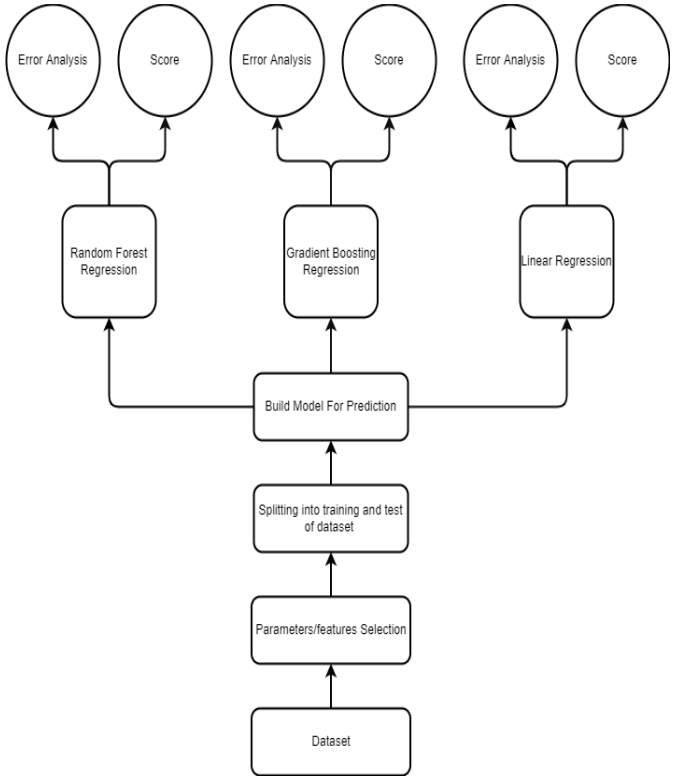

Figure 1: Generalized Model Building Approach

II. LITERATURE REVIEW

The related researches have been introduced in the recent years. These researches have addressed the issue for the individual one who does not know actual price of plot. Due to limited market insights, he/she might not sell/purchase their plot at low/high cost.

Sudhir N Dhage et al. [1] have addressed the issue about the price of houses in Mumbai city by applying the linear regression machine learning algorithms for prediction. The data-set was obtained from the government. The authors claimed that the difference between predicted prices and actual prices is 0.3713. Vivek Singh Rana et al. [2] have used support vector regression, XGBoost regression, decision tree regression, and random forest regression to predict the house prices. They considered major features such as location, size, society, availability, price, balcony, bath, and total sqft in the Bangalore city having 13320 samples (Kaggle data-set). They reported that decision tree model is highly prone to over-fitting (99% training data and 29% on testing data) whereas XGBoost found better, achieved 90% on training data and 63% on test. Sifei Lu et al. [3] have developed hybrid lasso, gradient boosting regression, and ridge regression models to predict individual house price. They have considered parameters such as location, size, house type, build year, and local amenities for prediction. These models have employed on as Kaggle competitions. They reported that hybrid regression produced 0.112 score (test data) by the combination of lasso regression (65%), and gradient boosting (35%). Neelam Shinde et al. [4] used the logistics regression, lasso regression, SVM, and decision tree regression machine learning algorithms for prediction of the house prices based on the physical parameters like location, area, material etc. in India. They have collected the data-set from the Kaggle which consisted of 3000 samples and 80 features. The authors have compared the models based on the performance metrices such as R^2, MAE, MSE, and RMSE. The reported that decision tree proved to be better than other algorithms having R^2 score of 0.99 and low error rates. The lasso achieved R^2 of 0.81, SVR (0.968), and LR (0.987). CH. Raga Madhuri et al. [5] developed house prediction models using MLR, ridge regression, lasso regression, elastic net regression, gradient boosting regression, and Ada boosting regression. They collected the data-set from the Vijayawada in India. They reported that gradient boosting regression algorithm-based model proved to be the best, achieving 0.91 accuracy. The Danh Phan in [6] used linear regression, polynomial regression, regression tree, neural network, step-wise & SVM, step-wise tuned SVM, PCA & SVM and PCA & tuned SVM for forecasting house prices in the Melbourne city of Australia. Data-set (Kaggle) which consisted of 34,857 samples and 21 features. The author has used step-wise, boosting, and PCA for data reduction and transformation. The author reported that the step-wise & tuned SVM has proved to be best, achieving R^2 score of 0.56 (0.0480 on trained MSE and 0.0561 on evaluation MSE).

III. DEVELOPED SYSTEM/METHOD

This section describes the adopted experimental method to carry out this work. There are six steps as shown below in figure 2.

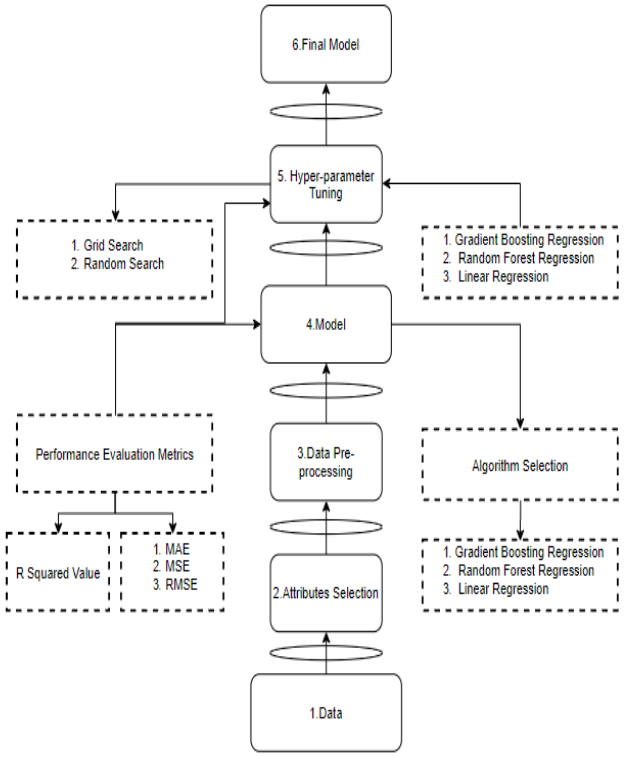

Figure 2:Plot price prediction system Architecture

A. Data-Collection

Data-set has been collected from Multan city in Pakistan. The data-set consists of two hundred eight samples that narrate 14 properties of a plot. Out of fourteen attributes, **Colony, Phase, Block, Zone, Size in Marla, Side, Park Facing, Market Facing, Near Mosque, Near School, Electricity, Natural Gas,** and **WASA** attributes are independent while **Price** is dependent variable. The data-set have all non-null values. The thirteen attributes of the data-set all have numerical values and remaining one attribute is of an object data type. The data-set attributes and their description are given in table 1.

Attributes	Description	Type
Colony	Attribute that describes the address of the street (name) where plot is located	Non-Numerical
Phase	Attribute that describes the number of phases in plot's area/town	Numerical
Block	Attribute that describes the number of blocks in that area/society	Numerical
Zone	Attribute that describes the number of zones in that area	Numerical
Size in Marla	Attribute that describes the size of plot in marla.	Numerical
Side	Attribute that describes the sides of the plot that are present.	Numerical
Park Facing	Attribute that describes the availability of park in that area	Numerical
Market Facing	Attribute that describes the availability of park in that area	Numerical
Near Mosque	Attribute that describes the availability of Mosque in that area	Numerical
Near School	Attribute that describes the availability of school in that area	Numerical
Electricity	Attribute that describes the availability of electricity for the plot in a society	Numerical
Natural Gas	Attribute that describes the availability of Natural Gas for the plot in a society	Numerical
WASA	Attribute that describes the availability of WASA for the plot in a society	Numerical
Price	Attribute that describes the price of the plot in that society	Numerical

Table 1:Feature Description Table

B. Data Pre-processing

The data-set pre-processing is an important process while building regression-based models. In the first instance, the independent variables are normalized using the Min-Max Scalar. Normalization scales down the input/function variables separately between the 0 and 1 range. The formula of Min-Max Scalar is:

$$z = \frac{(a - min)}{(max - min)} \quad (1)$$

Secondly, the normalized data-set is split into training and testing parts with the ratio of 8:2 respectively. Eventually, 166 records are used for the training and 42 entries are utilized for the testing purpose [7].

C. Machine Learning Algorithms

Three popular regression machine learning algorithms have been employed, for this work, to develop predictive models [7][5]. The description of these algorithms are as follows.

1) Multiple Linear Regression

Regression is divided into two parts simple linear and multiple regression. Multiple Linear Regressions is used to find relationship between a dependent variable and multiple independent variables [8]. The mathematical expressions are shown below.

$$y = mx + c \quad (2)$$

$$m = \frac{(y - y_1)}{(x - x_1)} \quad (3)$$

$$y = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 + \cdots + a_n a_n + \in \quad (4)$$

The equation (2) represents the hypothesis function for linear regression where 'm' represents that slope of the line whereas 'c' denotes the intercept. The equation (3) represents the slop equation. The equation (4) is the expanded form of equation (2) representing the multiple linear regression. Here 'a1', 'a2, and 'a3' are the coefficients of the line (combinedly representing the slope) and 'a0' is the intercept point.

2) Gradient Boosting

Gradient boosting is one of the stronger algorithm in ensemble learning. It is used in both classification and regression to solve the structuring predictive problems. The hypothetical theory narrates that it produces the better results when compared to the other ensemble models. It combines the weak learner into the strong learner [5]. Mathematically it is represented as:

$$Q_m = Q_{m-1}x + \alpha_1 * Q_m x \quad (5)$$

The equation (5) demonstrates the theoretical function of gradient boosting where 'Qm-1' represents the ensemble model while 'Qm' represents the weak learner. 'α' is the learning rate and 'x' represents the input vector.

3) Random Forest

Random forest is a supervised ensemble learning model which is employed for the classification and regression. its operation based on the decision tree [9]. It uses random data for training. Random forest use the bagging tree technique [8]. In random forest regression, decision tree works as a root/base learner and the output result is produced by the average prediction of individual trees. Random forest is the better one as compared single decision tree [18]. The mathematical representations of random forest regression are as:

$$f = \frac{1}{D}\sum_{d=1}^{D} f_p\,(y)' \quad (6)$$

$$\sigma = \sqrt{\sum_{d=1}^{D} \frac{(f_p(y)' - f')^2}{(D-1)}} \quad (7)$$

Where D is number of trees and y' is the error of prediction [10].

D. Hyper-Parameter Optimization Techinques

1) Grid Search

Grid Search is the most widely used method for tuning the hyper-parameters of the machine learning models for both classification and regression. In grid search, the specified set of values for each hyper-parameter is initialized by the user. The cartesian product of these user specified values sets are evaluated by the grid search. Grid search itself cannot exploit the well performing parameter region[11][12]. Such regions need to be defined manually by setting up lower and upper bounds for each hyper-parameter [11][12][13][14][15]. Here are some guiding principles for setting hyper-parameter regions:

- Start with the step sizes and large state search space.
- Squeeze the step sizes and the search space based on the results of the well preformed hyper-parameters that is previously generated and this work is repeated many times until optimal parameter is produced.

Grid search method is an effective technique for optimization. However, it has two major drawback; curse of dimensionality and high computational cost [11][14][15].

2) Random Search

Random search, a similar hyper-parameter tuning method to the grid search method. Random search resolves some limitations of grid search. Random search randomly chooses the specified values samples independently as the candidate hyper-parameters between the upper bounds and the lower bounds instead of performing brute force approach. With the limited resources, random search explores the larger space search. Global optimum or approximation can be achieved if the search space is large. It is faster than the grid search method and has still no knowledge for exploiting the well preforming region. The benefit of random search is that it samples the allotted number of parameters from distribution. This decreases the wasting much of the time on the poor performing region [11][14][15][16].

E. Performance Evaluation Metrics In Regression

The performance evaluation metrics are used to determine the range up to which a particular model's predictive results is distinctive from the actual results. In regression, R^2, MAE (Mean Absolute Error), MSE (Mean Squared Error), RMSE (Root Mean Squared Error) are popular means for evaluation. The later three calculate the residual for the model(s) [17].

1) R-Squared

R^2 is the measure of the proportion of variance that all independent variables in the model are used to explain in the dependent variable. The formula of the R^2 is given below:

$$R^2 = 1 - \frac{\mathrm{SS}_r}{\mathrm{SS}_t} \quad (8)$$

In above equation (8), 'SSr' is the sum of square residuals and the 'SSt' is the total sum of square of errors.

2) Mean Absolute Error

The equation of the Mean Absolute Error is:

$$MAE = \frac{1}{J}\sum_{a=1}^{A} |O_i - O| \quad (9)$$

Where J is the number of values, Oi is the predicted values, O is the actual values and (Oi-O) is the absolute errors[19].

3) Mean Squared Error

The Mean Square Error gives the square's errors average. The formula of the MSE is:

$$MSE = \frac{1}{J}\sum_{a=1}^{A} (O_i - O)^2 \quad (10)$$

In this equation, (Oi-O)2 square error[19].

4) Root Mean Squared Error

The square root of this Mean Squared Error equation provides the Root Mean Squared Error[19]. The RMSE is:

$$RMSE = \sqrt{\frac{1}{J}\sum_{a=1}^{A} (O_i - O)^2} \quad (11)$$

IV. RESULTS AND DISCUSSION

In this section, results of the experimental method are presented along with their discussion. The results produced from predictive models based on multiple linear regression, gradient boosting regression, and random forest regression algorithms are given below in the Table 2:

Algorithms	R^2	MAE	MSE	RMSE
Linear Regression	0.823	8.275716314100386	116.6325664212 6684	10.799655847353046
Gradient Boosting Regression	0.983	2.339239008330486	11.06693223763 5827	3.3266999019502537
Random Forest Regression	0.97	3.138772619047612	19.53074129422 612	4.419359828552787

Table 2:Table of Scores and Errors Results

In Table 2, the results for baseline models have been presented. As shown the R^2 score for gradient boosting regression is higher when compared to two other models. Also with MAE, MSE, and RMSE, gradient boosting regression performance is better in the base line model.

Table 3 shows the results produced through the models by optimization using grid search method:

Algorithms	R^2	MAE	MSE	RMSE
Linear Regression	0.823	8.275716314100393	116.6325664212 6701	10.799655847353055
Gradient Boosting Regression	0.987	1.858875285199403 7	8.800115933084 149	2.966498935291254 7
Random Forest Regression	0.971	3.138323214285724 5	18.92184488424 114	4.349924698686305

Table 3:Table of Scores and Errors Results by Grid Search Method

In Table 3, the results for the three developed models with hyper-tuning using grid search has been presented. As shown, the R^2 scores, MAE, MSE and RMSE for the gradient boosting regression is higher when compared to other models. However, the R^2 score of random forest regression is improved from the 0.97 to 0.971 and error rate like MAE is

also improved from 3.138772619047612 to 3.1383232142857245.

Table 4 shows the results produced through the models by optimization using random search method:

Algorithms	R^2	MAE	MSE	RMSE
Linear Regression	0.823	8.275716314100386	116.6325664212 6684	10.799655847353046
Gradient Boosting Regression	0.983	2.122667688688703 7	11.40534798656 9074	3.3771804788268387
Random Forest Regression	0.976	2.129741666666913	15.55611971261 9019	3.9441247080460093

Table 4:Table of Scores and Errors Results by Randomized Search Method

In Table 4, the results for these models with hyper-parameter tuning using random search have been presented. Although the improvement of R^2 and error rate such as MAE in random forest regression from 0.97 to 0.976 and from 3.138772619047612 to 2.1226676886887037, the improved gradient boosting regression is yet better when compared to other models.

A. Quantitative Results Comparison

Table 5 displays the quantitative comparison of related research work. As shown multiple linear regression algorithm in [5] obtained R^2 score of 0.732 and in [20] obtained 0.41 R^2 score whereas our work achieved R^2 score of 0.823. Gradient boosting regression in [5] got R^2 score of 0.917 while our work accomplished 0.983 R^2 score. Random forest regression in [9] produced R^2 score of 0.8019 whereas our work attained R^2 score 0.97 .

B. Graphical Representation of Results

The scatter-plots shown here represent the relationship between the predicted price from the independent variables and actual price of the plot. Since the divergence of points is very high from the linear line, therefore, multiple linear regression proves to be worst efficient as shown graphically in Figure 3:

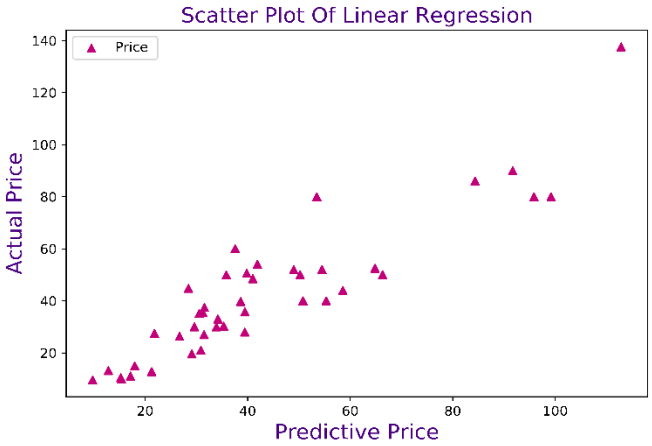

Figure 3:Scatter Plot of Linear Regression

The figure 4 represents that the points drawn from the scatter plot constitute that the actual price and predictive price forms the linear line while using gradient boosting regression algorithm. So, the divergence of the points of the gradient boosting regression is least proving highly efficient method.

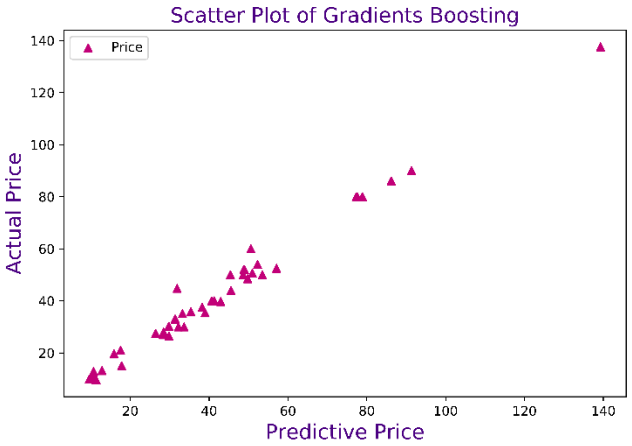

Figure 4:Scatter Plot of Gradient Boosting Regression

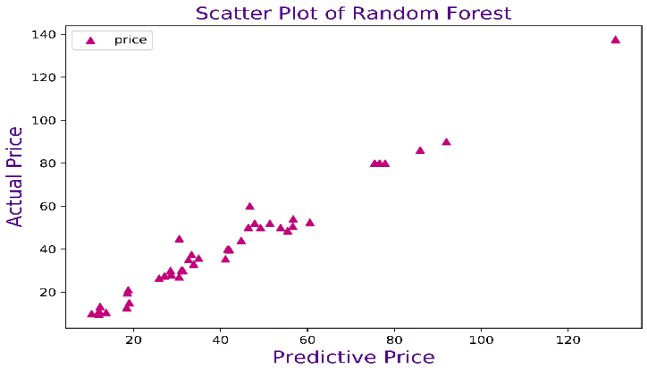

Figure 5:Scatter Plot of Random Forest Regression

Ref.	Algorithm	Score		MAE		MSE		RMSE	
		Other Work	This Work	Other	This Work	Other Work	This Work	Other Work	This Work
[5]	MLR	0.732	0.823	-	8.27571631410039 3	39187574448.88446	116.6325664212670 1	19795851699	10.799655847353055
[5]	GBR	0.917	0.983	-	2.33923900833048 6	12037006088.27804	11.06693223763582 7	10971390390	3.32669990195025 37
[20]	MLR	0.41	0.823	-	8.27571631410039 3	---	116.6325664212670 1	0.0912	10.799655847353055
[9]	RFR	0.8019	0.97	-	3.13877261904761 2	---	19.5307412942261 2	95928.32	4.41935982855278 7

Table 5:Comparative Studies of Results

A high divergence of the points, as shown in figure 5, demonstrates that model using random forest regression algorithm is the less efficient method comparative to multiple linear regression method.

The Bar-plots have been used in this work for plotting the comparative R^2 score of baseline models. The bar plot in figure 6 shows that on comparison of R^2 score of baseline models, gradient boosting regression is performed better compared to other models.

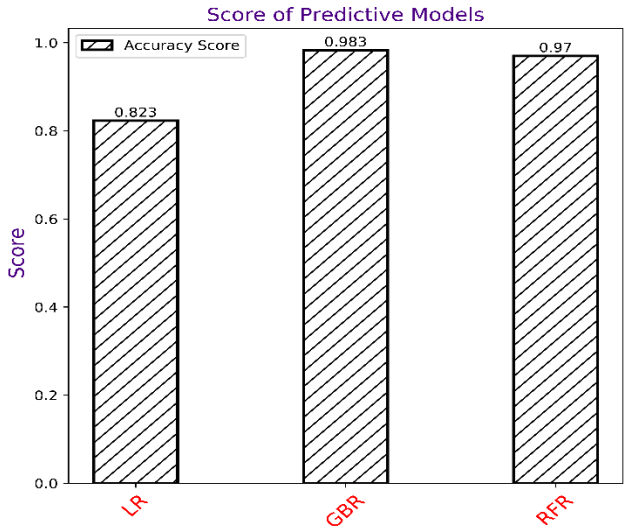

Figure 6:Score Comparison of Models

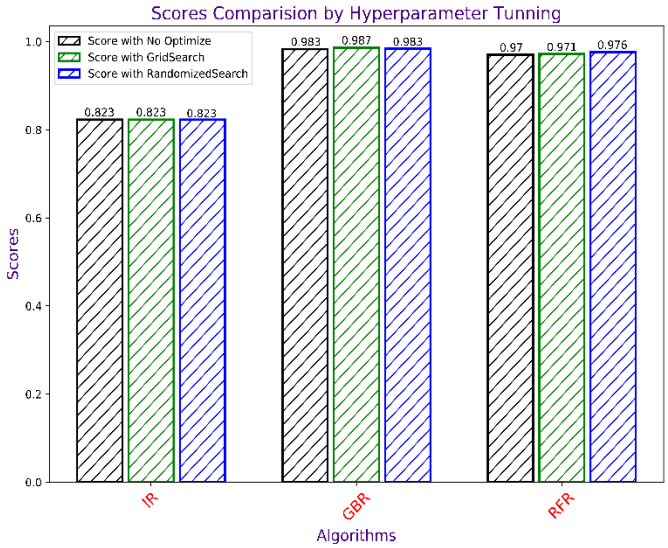

Figure 7:Scores Comparison of Models by optimization using hyper-parameter tuning

The figure 7 shows that by plotting the bar graph of comparative R^2 score of baseline models without optimization and with optimization methods that gradient boosting regression performed better compared to others. The other baseline model such as random forest regression is improved well with optimization methods.

V. CONCLUSION

Common people use his/her life long savings to purchase a plot. The savings are prone to be robbed by market fraudulents. Therefore, a computer-based market price prediction/assessment system is highly required to provide reliable estimated prices to them. This work used Multan City plotting data-set which is the well-known city in Pakistan. The solution was proposed for this problem by applying three

machine algorithms without and with hyper-parameter tuning using grid search and random search methods. The results of these models were compared by R^2. It was found from the results that gradient boosting regression algorithm proved to be the best algorithm as a baseline method to implement a model. The obtained results also demonstrated that the same technique is effective after applying optimization, achieving 0.987 R^2 score and lower error values of metrics such as MAE, MSE, and RMSE. The multiple linear regression algorithm-based model achieved 0.823 R^2 score and higher error values. For future work, it is recommended that a new data-set fulfilling local requirements may be used.

REFERENCES

[1] N. N. Ghosalkar and S. N. Dhage, "Real Estate Value Prediction Using Linear Regression," *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018*, 2018, doi: 10.1109/ICCUBEA.2018.8697639.

[2] V. S. Rana, J. Mondal, A. Sharma, and I. Kashyap, "House Price Prediction Using Optimal Regression Techniques," *Proc. - IEEE 2020 2nd Int. Conf. Adv. Comput. Commun. Control Networking, ICACCCN 2020*, pp. 203–208, 2020, doi: 10.1109/ICACCCN51052.2020.9362864.

[3] S. Lu, Z. Li, Z. Qin, X. Yang, and R. S. M. Goh, "A hybrid regression technique for house prices prediction," *IEEE Int. Conf. Ind. Eng. Eng. Manag.*, vol. 2017-Decem, pp. 319–323, 2018, doi: 10.1109/IEEM.2017.8289904.

[4] N. Shinde and K. Gawande, "Valuation of House Prices Using Predictive Techniques," *Int. J. Adv. Electron. Comput. Sci.*, vol. 5, no. 6, pp. 34–40, 2018, [Online]. Available: http://iraj.in.

[5] C. H. R. Madhuri, G. Anuradha, and M. V. Pujitha, "House Price Prediction Using Regression Techniques: A Comparative Study," *6th IEEE Int. Conf. "Smart Struct. Syst. ICSSS 2019*, pp. 4–8, 2019, doi: 10.1109/ICSSS.2019.8882834.

[6] T. D. Phan, "Housing price prediction using machine learning algorithms: The case of Melbourne city, Australia," *Proc. - Int. Conf. Mach. Learn. Data Eng. iCMLDE 2018*, pp. 8–13, 2019, doi: 10.1109/iCMLDE.2018.00017.

[7] G. Srirutchataboon, S. Prasertthum, E. Chuangsuwanich, P. N. Pratanwanich, and C. Ratanamahatana, "Stacking Ensemble Learning for Housing Price Prediction: A Case Study in Thailand," *KST 2021 - 2021 13th Int. Conf. Knowl. Smart Technol.*, pp. 73–77, 2021, doi: 10.1109/KST51265.2021.9415771.

[8] A. Varma, A. Sarma, S. Doshi, and R. Nair, "House Price Prediction Using Machine Learning and Neural Networks," *Proc. Int. Conf. Inven. Commun. Comput. Technol. ICICCT 2018*, pp. 1936–1939, 2018, doi: 10.1109/ICICCT.2018.8473231.

[9] T. Oladunni, S. Sharma, and R. Tiwang, "A spatio-temporal hedonic house regression model," *Proc. - 16th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2017*, vol. 2017-Decem, pp. 607–612, 2017, doi: 10.1109/ICMLA.2017.00-94.

[10] J. Wu, X. Y. Chen, H. Zhang, L. D. Xiong, H. Lei, and S. H. Deng, "Hyperparameter optimization for machine learning models based on Bayesian optimization," *J. Electron. Sci. Technol.*, vol. 17, no. 1, pp. 26–40, 2019, doi: 10.11989/JEST.1674-862X.80904120.

[11] T. Gressling, *84 Automated machine learning*. 2020.

[12] M. N. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, "Systematic ensemble model selection approach for educational data mining," *Knowledge-Based Syst.*, vol. 200, p. 105992, 2020, doi: 10.1016/j.knosys.2020.105992.

[13] D. Sangani, K. Erickson, and M. Al Hasan, "Predicting Zillow Estimation Error Using Linear Regression and Gradient Boosting," *Proc. - 14th IEEE Int. Conf. Mob. Ad Hoc Sens. Syst. MASS 2017*, pp. 530–534, 2017, doi: 10.1109/MASS.2017.88.

[14] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020, doi: 10.1016/j.neucom.2020.07.061.

[15] M. A. Zöller and M. F. Huber, "Benchmark and Survey of Automated Machine Learning Frameworks," *J. Artif. Intell. Res.*, vol. 70, pp. 409–472, 2021, doi: 10.1613/JAIR.1.11854.

[16] R. C. Intelligence, "REIHE COMPUTATIONAL INTELLIGENCE by Simple Randomized Search Heuristics," no. September, 2004.

[17] N. Nghiep and C. Al, "Predicting Housing Value: A Comparison of Multiple Regression Analysis and Artificial Neural Networks," *J. Real Estate Res.*, vol. 22, no. 3, pp. 313–336, 2001, doi: 10.1080/10835547.2001.12091068.

[18] B. Park and J. Kwon Bae, "Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data," *Expert Syst. Appl.*, vol. 42, no. 6, pp. 2928–2934, 2015, doi: 10.1016/j.eswa.2014.11.040.

[19] W. K. O. Ho, B. Tang, and S. W. Wong, "Predicting property prices with machine learning algorithms," *J. Prop. Res.*, vol. 38, no. 1, pp. 48–70, 2021, doi: 10.1080/09599916.2020.1832558.

[20] J. Manasa, R. Gupta, and N. S. Narahari, "Machine Learning based Predicting House Prices using Regression Techniques," *2nd Int. Conf. Innov. Mech. Ind. Appl. ICIMIA 2020 - Conf. Proc.*, no. Icimia, pp. 624–630, 2020, doi: 10.1109/ICIMIA48430.2020.9074952.